

IN THE CLAIMS:

All pending claims are produced below.

1. (Currently Amended) A method of protecting application program software
executing API code including:

actuating a tracer function to copy a segment of instructions from the API code until
an instruction from the API code is reached that is one selected from the group consisting of
a call instruction ending outside of scope, a jmp instruction ending outside of scope, a
sysenter instruction, a syscall instruction, and a branch instruction ending outside of scope, or
until an instruction above a predetermined number of instructions is reached, wherein the
predetermined number of instructions is above two instructions;

storing and executing the copied instructions; and

returning to the next instruction of the API code, wherein the next instruction of the
API code is a first uncopied instruction of the API code.

2. (Previously Presented) The method of claim 1 wherein the segment of instructions is
a maximum of 16 instructions and the copied instructions are stored in the Random Access
Memory (RAM) of the CPU.

3. (Currently amended) A The method ~~as claimed in~~ of claim 1 wherein the application
program software is security program software.

4. (Currently amended) A The method of ~~protecting application program software as~~
~~claimed in~~ claim 1 wherein the tracer function includes the following instructions:

read instruction of *myfunction* (interpret opcodes);

if instruction is *not* ((a call, jmp, sysenter, syscall or branch instruction which
ends up out of scope) or (less than the 16th instruction)) then copy to the local buffer;

repeat above steps until out of scope;

execute the local buffer;

continue execution in the original *myfunction* code at the offset where the out of scope instruction was encountered.